On Using UML in the Systems Engineering Process for System Requirements Development

Pete Ross, Shawn Simmons, Michael Crow

The Boeing Company

Integrated Defense Systems

BOEING®

1.0 Introduction

1.1 Need for Improved Methods of Implementing SE Process

1.2 Where OOA, UML, and Related Tools Help

2.0 Comprehensive Requirements Analysis and Management Through Models & Databases

2.1 UML Use for Enhanced Definition & Communication of Requirements

2.2 Executable Modeling for Requirements & Design Analyses

3.0 Summary

BOEING®

1.1 Need for Improved Methods of Implementing SE Process

BOEING®

Large Scale Platforms Containing Highly Integrated Systems

BOEING®

Many Platforms, Highly Integrated

Integrated Air Platforms

Integrated Air And Ground Systems

Integrated Ground Platforms

Common Operating Picture And Information Available At All Echelons

Highly Integrated Systems Engineering Processes are Needed to Meet The Challenges Of Large Scale Integration Of System of Systems

BOEING®

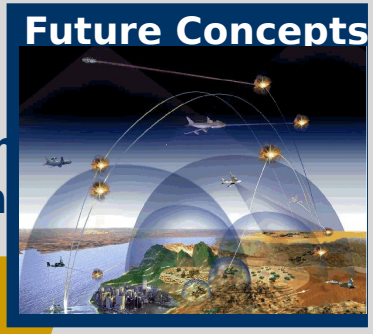
The System Engineering Environment Has Changed

Operational Needs Are Continuously Changing

Systems Technologies are Constantly Evolving

Therefore Requirements Evolve Continuously during A Development Program

Future
- **System-of-Systems Solution For Network Centric Opera**
- **Flexibility/Agility**
- **"Time to Market" Focus**
- **Commercial Practices and Products**
- **MDA/ Effects Based Development**
- ***Quality at Each Step of Process***



Present
- **Integrated Systems**
- **Rapid Prototyping**
- **COTS**
- **Affordability Focus**

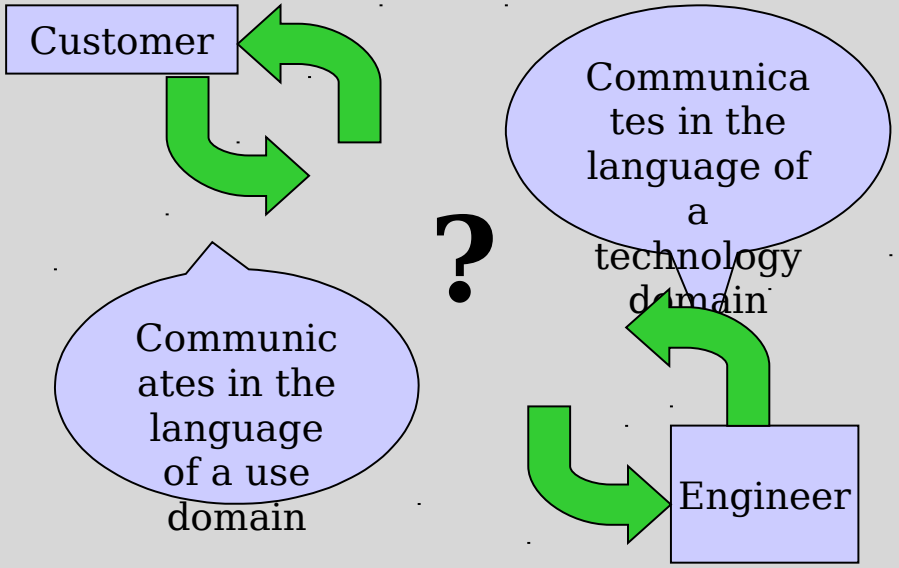
Past
- **Performance Driven**
- **Military Specs**
- **Federated Systems**
- **Focus on Delivered Quality**

Systems Engineering Processes Must Support Flexibility Throughout Development Process

BOEING®

The Communication Challenge

- **Capture and Integrate Operator Intended Usage to Properly Turn User Needs into System Requirements**
  - Need to keep customer in the loop well into the development process
  - For efficiency and clarity, need to be able to effectively communicate with the intended users in a language that both they **and** the contractor team understand



- **Capture and Integrate Analyses of Many Domain Experts to Properly Turn User Needs into System Requirements**
  - Increased emphasis on well organized and maintainable capture and analysis of user and inter-/ intra- system processes in a variety of situations
  - For efficiency and clarity, need integrated processes, modeling languages, and tools
- **Address Legacy Structured Analysis and Design Technique Shortfalls**
  - Tends to drive federated solutions
  - Low support for showing linkage of functional decomposition back to intended use

BOEING®

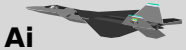
The Systems Integration Challenge For SoS Architectures

Increased Focus of Integration at the Highest Level and There
Complexity From Both the Operators and Developers Point of

Force Structure Level 1(Context)
Requirements Detail Allocates to System

JTF/AOC

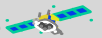
Air



Land

Sea

Space



Theater Command & Control Processes/Interactions

Users Operational Vie

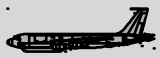
DOD AF Intent is to Bridge these Levels

System Developers Vie

Functional & Physical Architectures

Weapon System Level 2 (Concept)
Allocates to Segment

Air Vehicle S

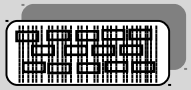


ICD

Control Segment

ICD

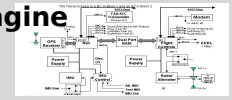
Radar Segment



AV ICD:

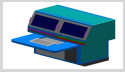
Major Subsystem Level 3 (Logical)
Allocates to SubSys

eg. Cockpit, airframe, engine



Internal ICDs

eg. Console, Comm gear



Internal ICDs

Integrated Processor

Subsystem* Level 4 *Allocates to CSCI/HWCI

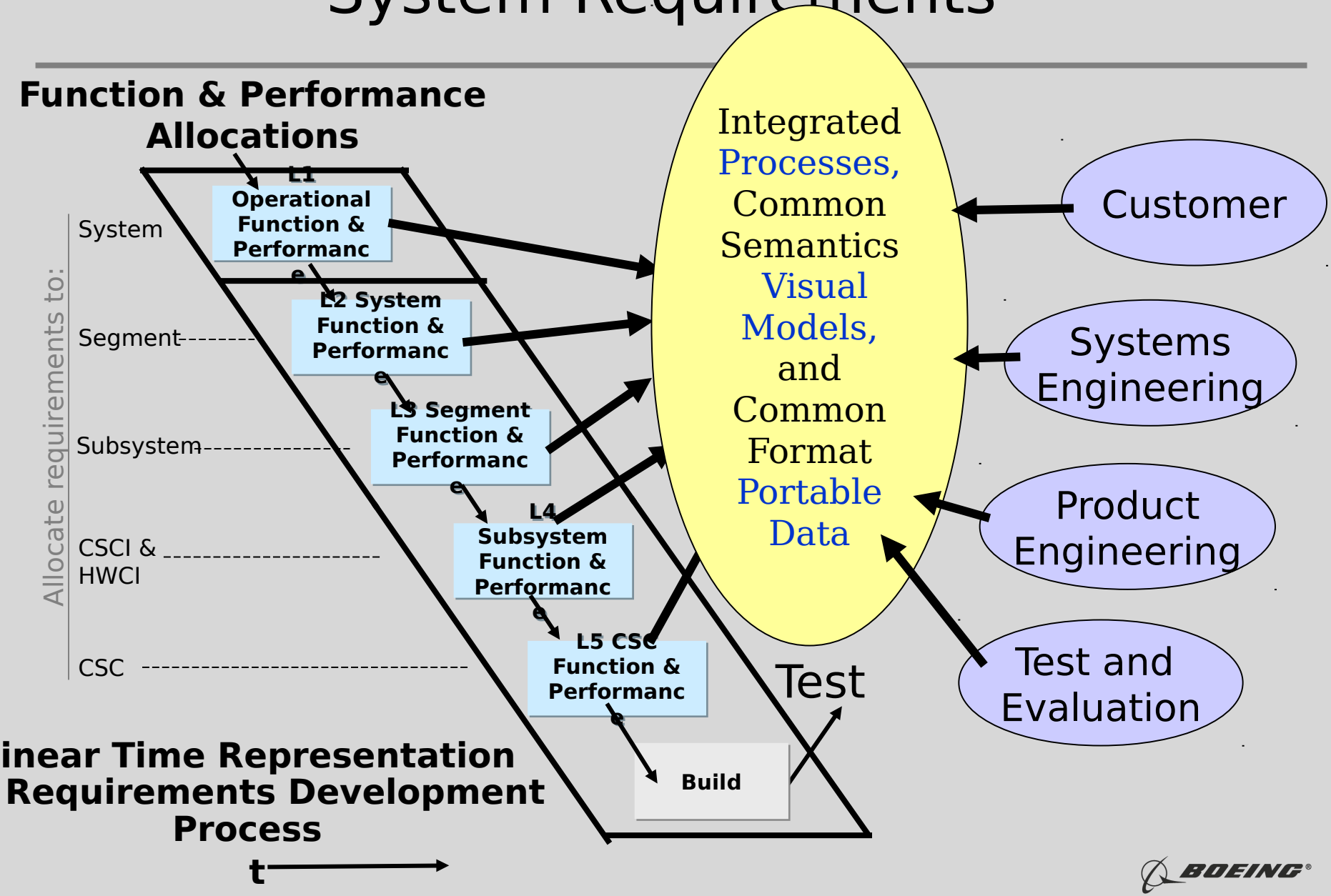
***CI & CSCI* Level 5 Allocates to HWC and CSC**

Integration Has Been Focused At This Level Over Last Decade

Traditional SE Processes Have Not Been Proven To Be Efficient

BOEING®

Definition of System Requirements



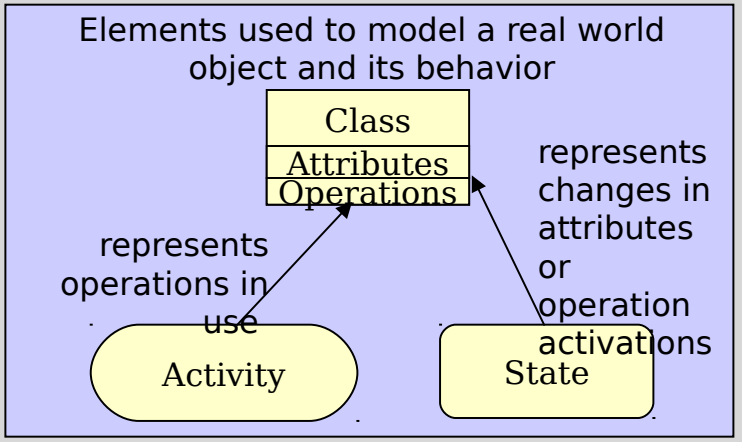
BOEING®

1.2 Where OOA, UML, and Related Tools Help

BOEING®

OOA/ UML Benefits

Captures Life Cycle Uses of System By Formal Analysis of Multiple Use Cases



Integrates Abstract Physical & Behavioral Views

Using Object Oriented Analysis

Commercial and defense industry software development standard modeling language

UML is a standardized language Which has Continued software and systems engineering support

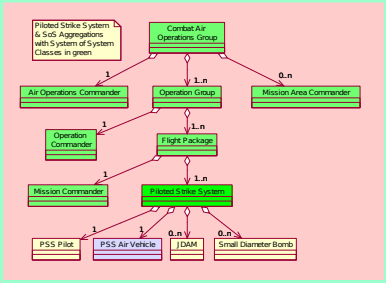
BOEING®

What Are Object Oriented Analysis and Unified Modeling Language?

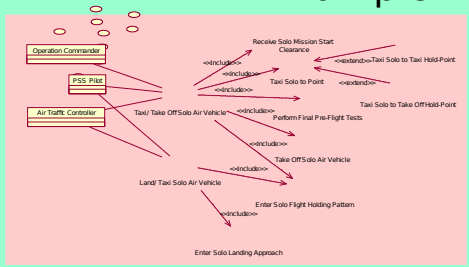
- OOA: Abstract modeling and analysis of real world objects behavioral interactions and architectural relationships
  - Apply Content Decomposition Hierarchically to Aid in System Requirements Analysis
  - OO is Used to Bound the Levels and Support Information Technology Based Descriptions of the System
- UML: UML captures the results of OOA
  - UML is an industry standard for conceptual/ visual modeling
  - Specified and maintained by non-profit Object Management Group (OMG)
  - UML is a language not a methodology
    - Concepts, Semantics, Notation, Rules

Diagram/View Types: Activity, Class, Collaboration, Component, Deployment, Sequence, State, Use

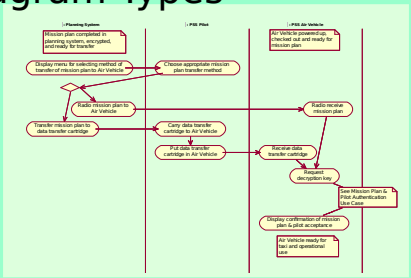
Example Diagram Types



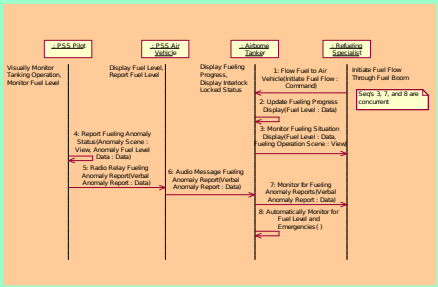
Class Diagrams


Operation Commander
PSS Pilot
Air Traffic Controller
Receive Solo Mission Start Clearance
Taxi Solo to Point
Taxi Solo to Taxi Hold Point
Taxi Solo to Take Off Hold Point
Taxi/ Take Off Solo Air Vehicle
Perform Final Pre-Flight Tests
Take Off Solo Air Vehicle
Land/ Taxi Solo Air Vehicle
Enter Solo Flight Holding Pattern
Enter Solo Landing Approach

Use Case Diagrams



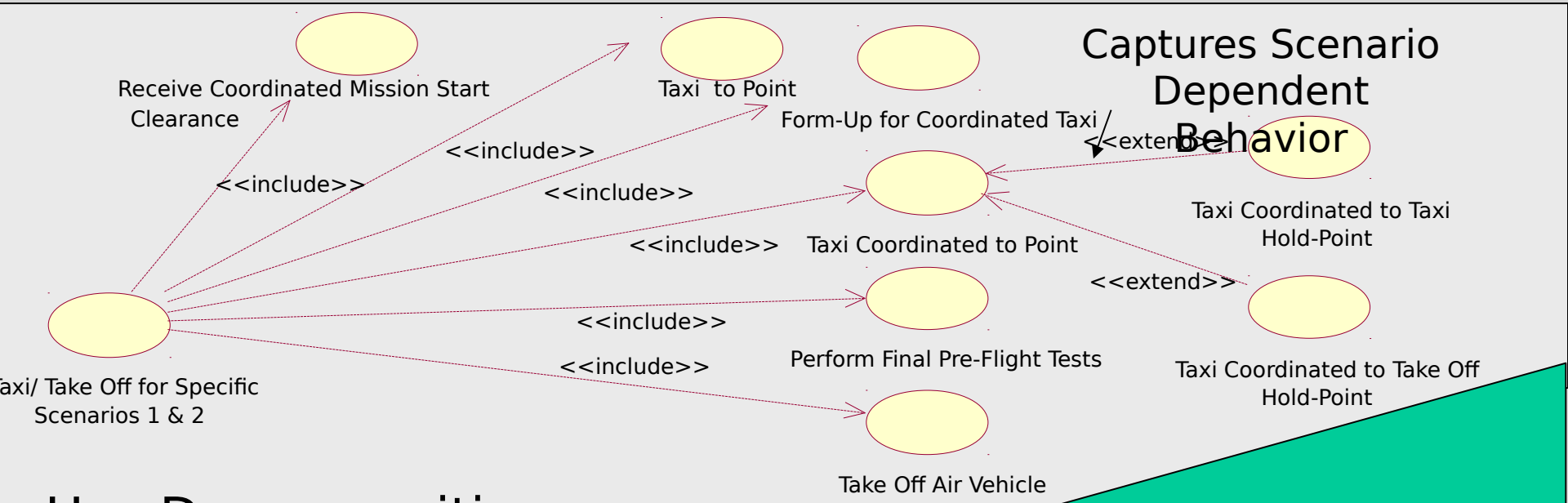
Activity & State Diagrams


: PSS Pilot
: PSS Air Vehicle
: Airborne Tanker
: Refueling Specialist
Visually Monitor Tanking Operation, Monitor Fuel Level
Display Fuel Level, Report Fuel Level
Display Fueling Progress, Display Interlock Locked Status
1: Flow Fuel to Air Vehicle(Initiate Fuel Flow : Command)
Initiate Fuel Flow Through Fuel Boom
2: Update Fueling Progress Display(Fuel Level : Data)
3: Monitor Fueling Situation Display(Fuel Level : Data, Fueling Operation Scene : View)
4: Report Fueling Anomaly Status(Anomaly Scene : View, Anomaly Fuel Level Data : Data)
5: Radio Relay Fueling Anomaly Report(Verbal Anomaly Report : Data)
6: Audio Message Fueling Anomaly Report(Verbal Anomaly Report : Data)
7: Monitor for Fueling Anomaly Reports(Verbal Anomaly Report : Data)
8: Automatically Monitor for Fuel Level and Emergencies( )

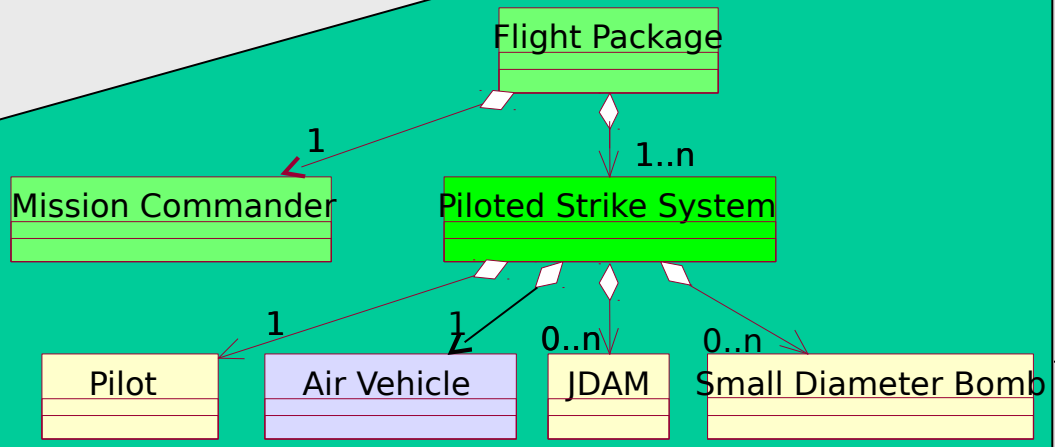
Sequence Diagrams

BOEING®

OOA/ UML Supports Decomposition of Use, Behavior, Objects, Data, and Interface Types



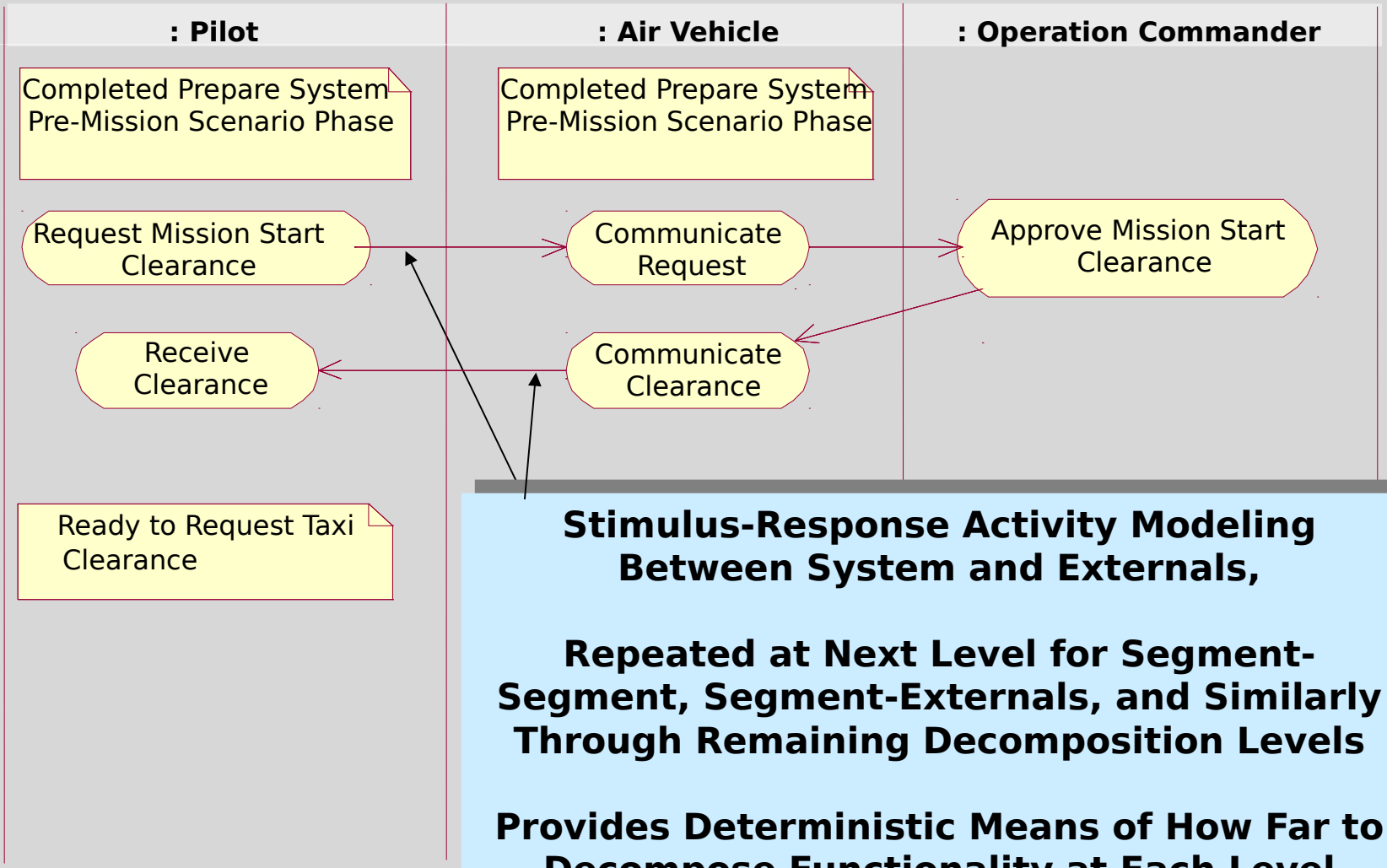
Use Decomposition



Content Decomposition

BOEING®

OOA/ UML Used to Capture Interface Behavior One Level at a Time



Stimulus-Response Activity Modeling Between System and Externals,

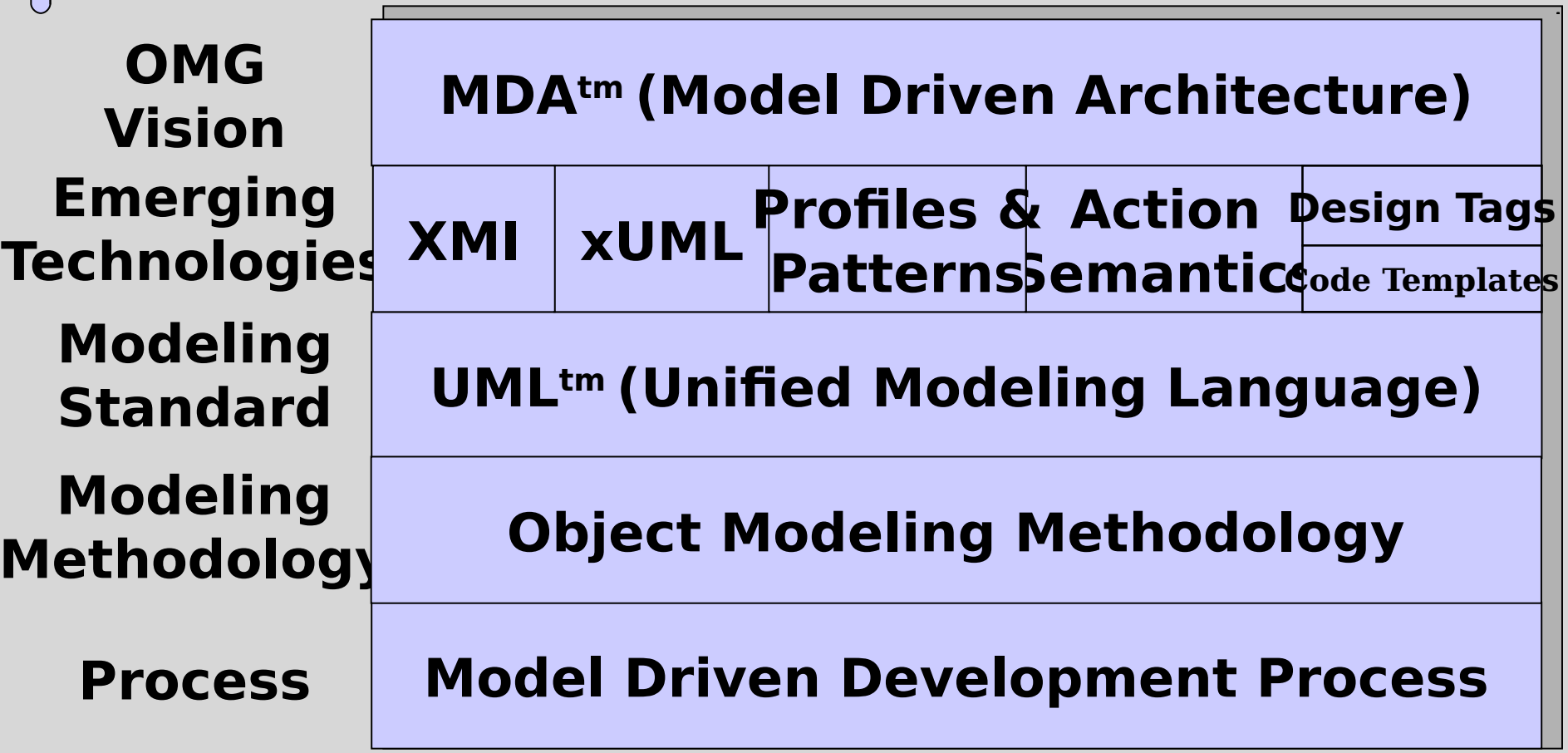
Repeated at Next Level for Segment-Segment, Segment-Externals, and Similarly Through Remaining Decomposition Levels

Provides Deterministic Means of How Far to Decompose Functionality at Each Level

BOEING®

UML Supporting Technologies

MG is working to provide a vision, framework, and definition at all levels software and systems development on a non-proprietary basis



BOEING®

What is needed beyond OOA/ UML?

- **Integration With Performance Requirements Analysis Tools & Processes**
  - An adequate means to capture and manage performance requirements
  - Need visual models to capture timing and motion requirements
  - Need visual models for performance parameter decomposition/ allocation
  - Adequate functional architecture views for use as a framework for trade-offs of physical architectures and performance allocation analyses
  - Adequate semantics for combined behavioral and performance simulations
    - xUML provides some capability in this area
- **Integration With Physical Modeling Tools & Processes**
  - Adequate physical views for capturing and managing physical interfaces, physical allocation analyses, and non-functional requirements
- **Integration With Classical System Engineering Processes**
  - Integrated cost, performance, schedule, risk models for use in trade-off analyses

BOEING®

2.0 Comprehensive Requirements Analysis and Management Through Models & Databases

2.1 UML Use for Enhanced Definition & Communication of Requirements

2.2 Executable Modeling for Requirements and Design Analyses

BOEING®

2.1 UML Used for Enhanced Definition & Communication of Requirements

BOEING®

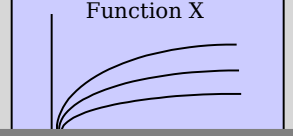
The Coordination Challenge

Stakeholders

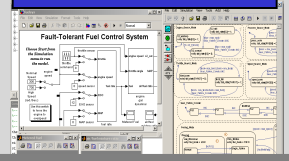
- Customer
- Product Engineering
- Test and Evaluation
- Systems Engineering

Integrated Processes, Common Semantics
Visual Models, and Common Format Portable Data

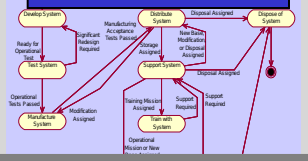
- Matrices/ Databases
- Graphs



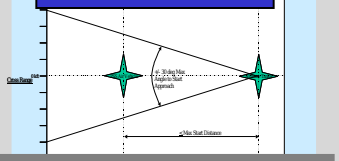
- Simulations



- Block Diagrams



- Drawings

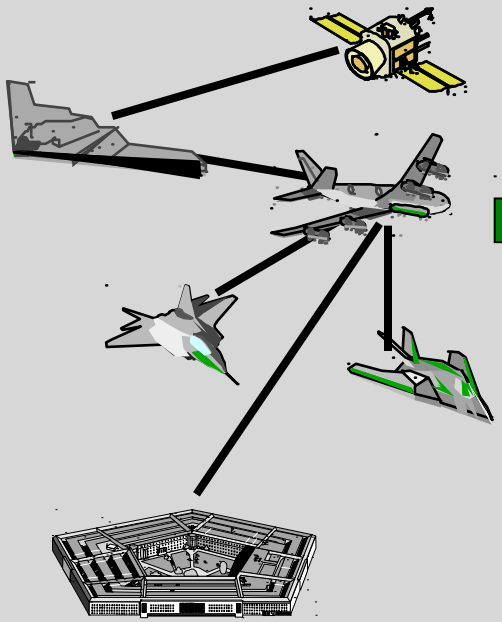


Map Best Models & Data Constructs to an Integrated SE Process
Use UML Model as Focal Point to Tie Together Engineering Products

BOEING®

Multi-Discipline Participation in Use Case Driven Architecture Development & Design

Interoperability



System Con Ops, Safety, Security

Architecturally Significant Mission Scenarios

System Effectiveness Reliability

Measures of Effectiveness

Nominal Use Threads
Contingency Threads

Abstract Use Case "Operate System"

Performance Requirements Database

UML Model

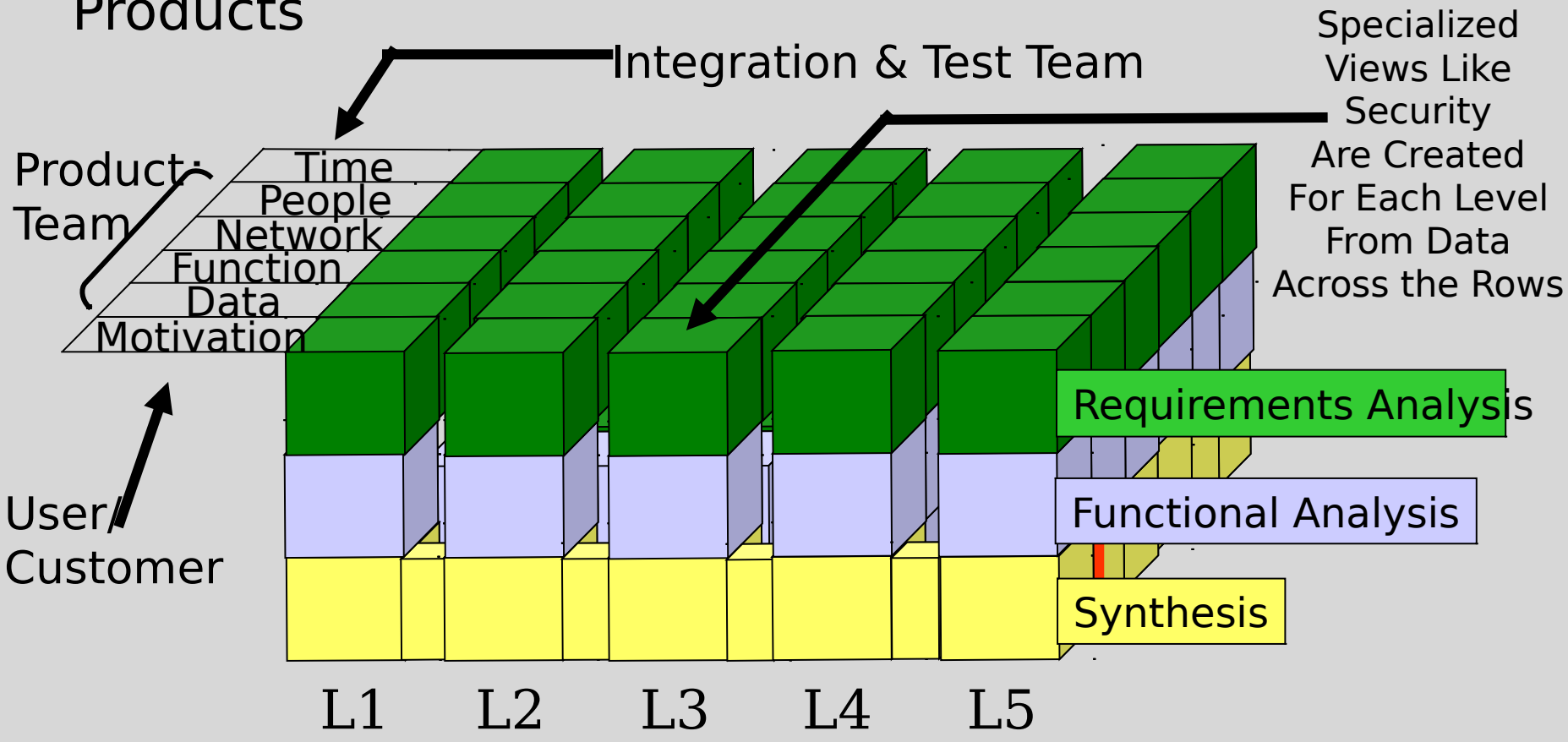
Context Diagrams, Process Flows, Use Descriptions, Item Exchanges, Candidate Technology &Solution Objects Classes

Architecturally Significant Mission Scenarios Identify Functional Interaction of System with External Participants And Use of IERs

BOEING®

Views Need to Convey All Aspects of Each Level Of The Architecture To All Stakeholders

Cubes Represent Engineering Data & Architecture Products



The Products Developed are Organized Based on the Zachman Framework
The goal of Using a Framework is to Ensure an Architecturally Complete Description

BOEING®

Engineering Products Provide Forum and Canvas for Product Definition and Refinement

Architecturally Significant Mission Scenarios

Measures of Effectiveness

Capabilities

External Systems

Use Cases
(for all life cycle phases)

Modes

States

Threads

Activities

Views

Product Relationships Provide Linkage To Intended Use Scenarios

Database

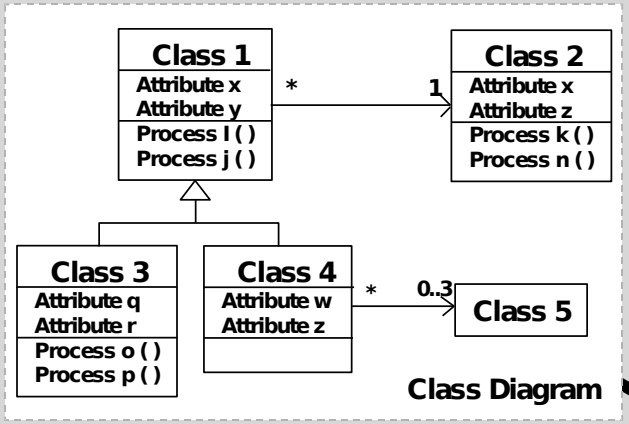
Performance

Functional Architecture

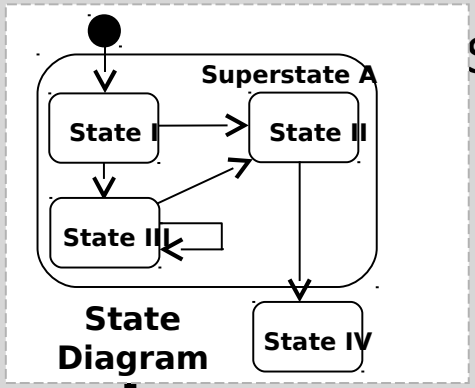
Physical Architecture

BOEING®

Construction of The Functional Architecture And Allocated To The Physical Architecture

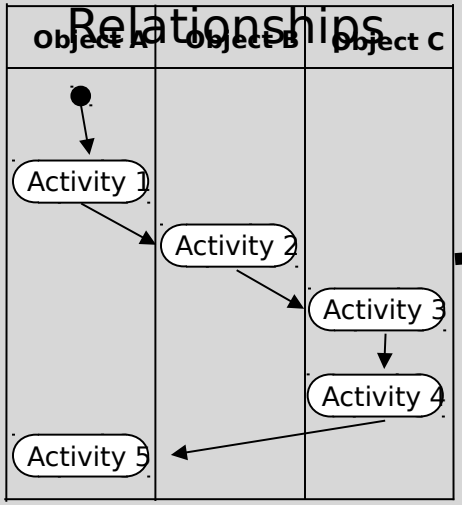


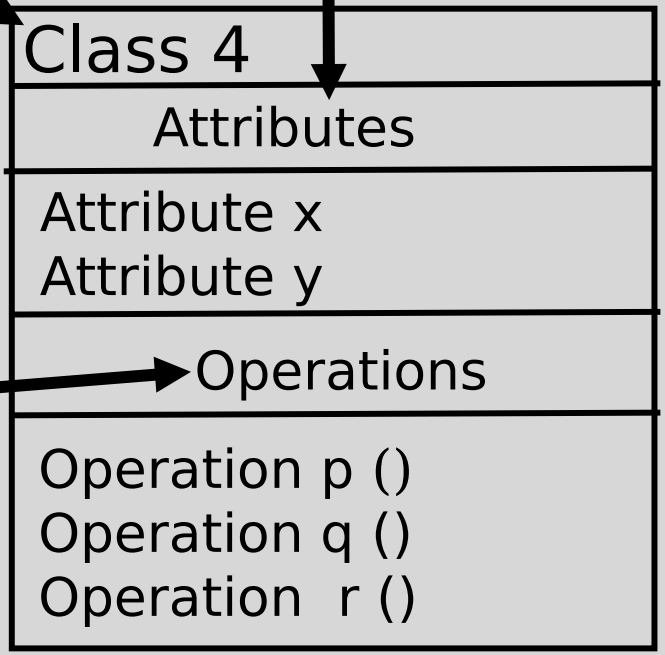
Class Diagram

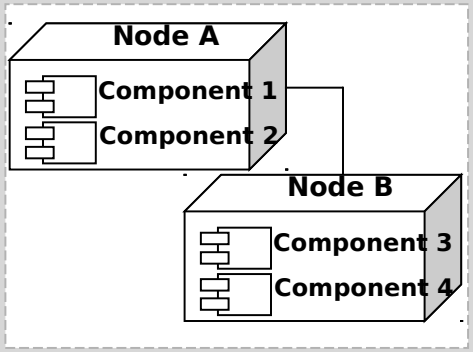


State Diagram

Class Association (roles) Relationships







Deployment Diagram

BOEING®

Integrated Process & Environment for Deriving Behavioral Requirements

Use Driven and Methodical Requirements Derivation

Capabilities

Life Cycle Phases

External Systems

Safety, Health, Quality, Security, Survivability, Concepts

Use Cases

UML

Threads

Activities

Modes

States

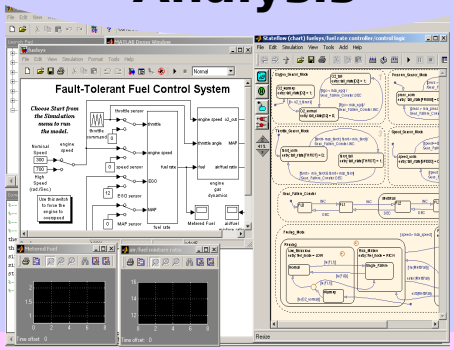
Matrix

Functional Architecture

IDEF

Performance

Performance Analysis



Use Scenario Script Driven Verification Using Logic and Timing Simulations

State Model

Performance Model

Interactive Simulation

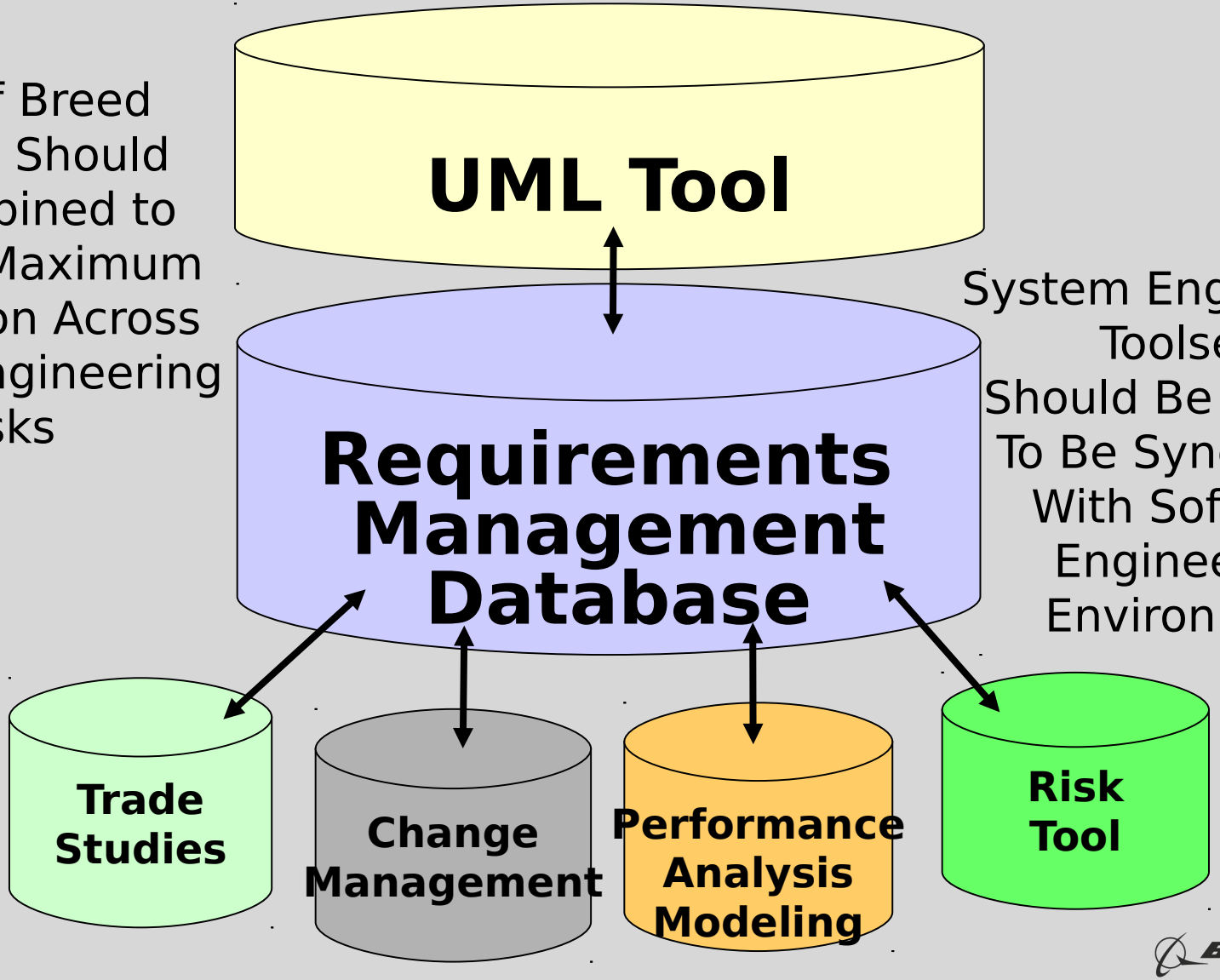
Linking performance with functional architecture to validate

Architecture meets Measures of Effectiveness is key

BOEING®

Integrate Commercially Available Tools

Best Of Breed Toolsets Should Be Combined to Achieve Maximum Integration Across System Engineering Tasks



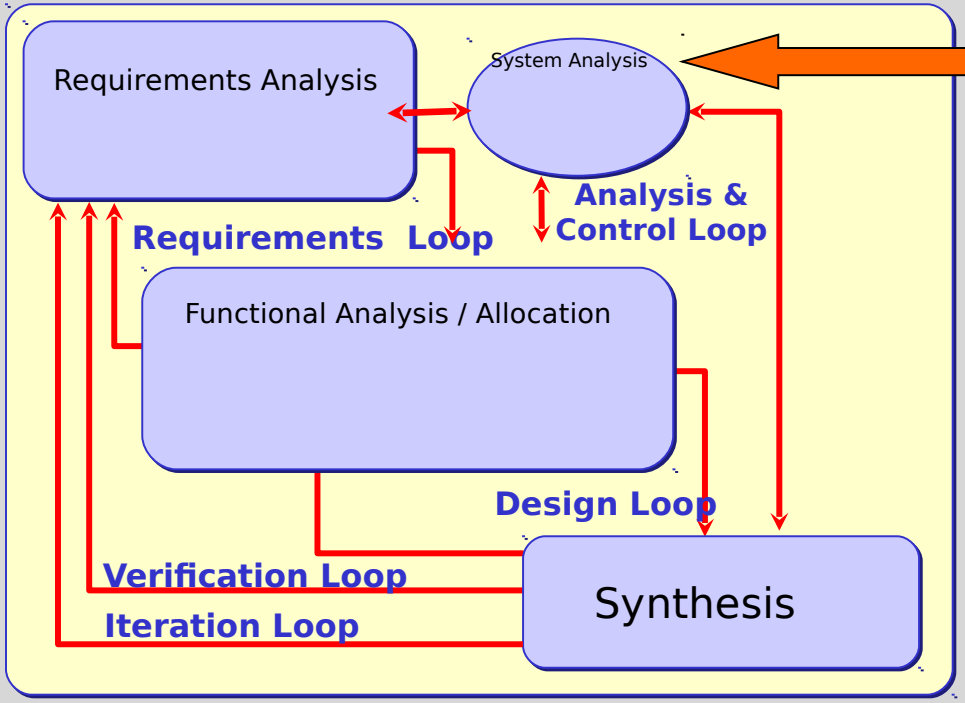
System Engineering Toolsets Should Be Chosen To Be Synergistic With Software Engineering Environment

BOEING®

2.2 Executable Modeling for Requirements and Design Analyses

BOEING®

Where Executable Modeling Fits In



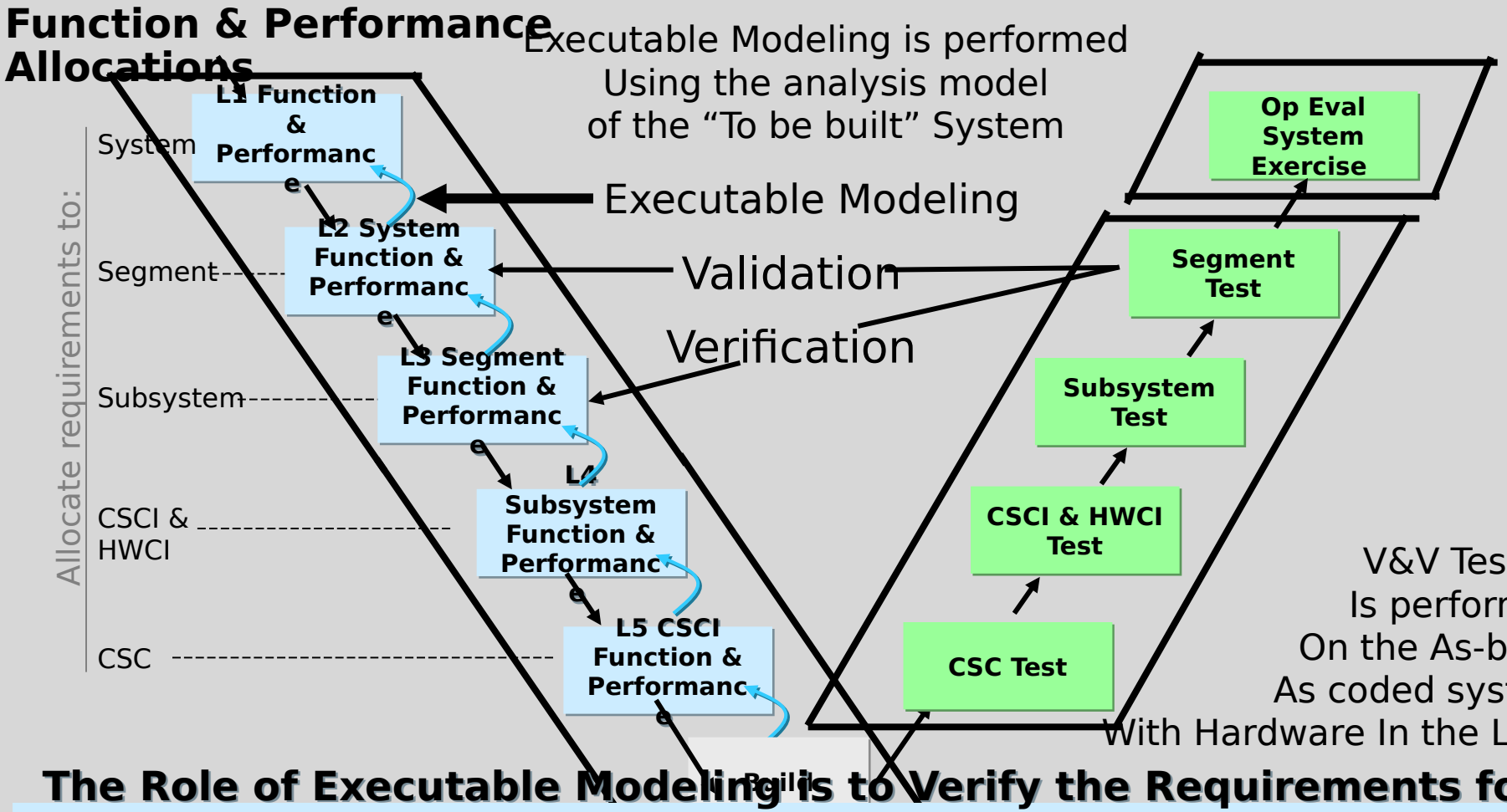
In the Development Proces

- To Meet System Analyses Needs:
  - **Capacity (bandwidth)**
  - **Security**
  - **Fault Tolerance**
  - **Interoperability**
  - **Interference**
  - **Operator Awareness**
  - **Control Logic, Timing, Accuracy**
  - **Safety (Maintain Physical Separation)**
  - **Timelines and accuracies required to Carry out Mission Objectives**

- Supports SE Information Needs:
  - Critical Function Partitioning
  - Data for Systems Engineering Trade Studies and “what if” analysis.
  - Data to Verify requirements and architecture completeness.
  - Data to Verify interface definition.

BOEING®

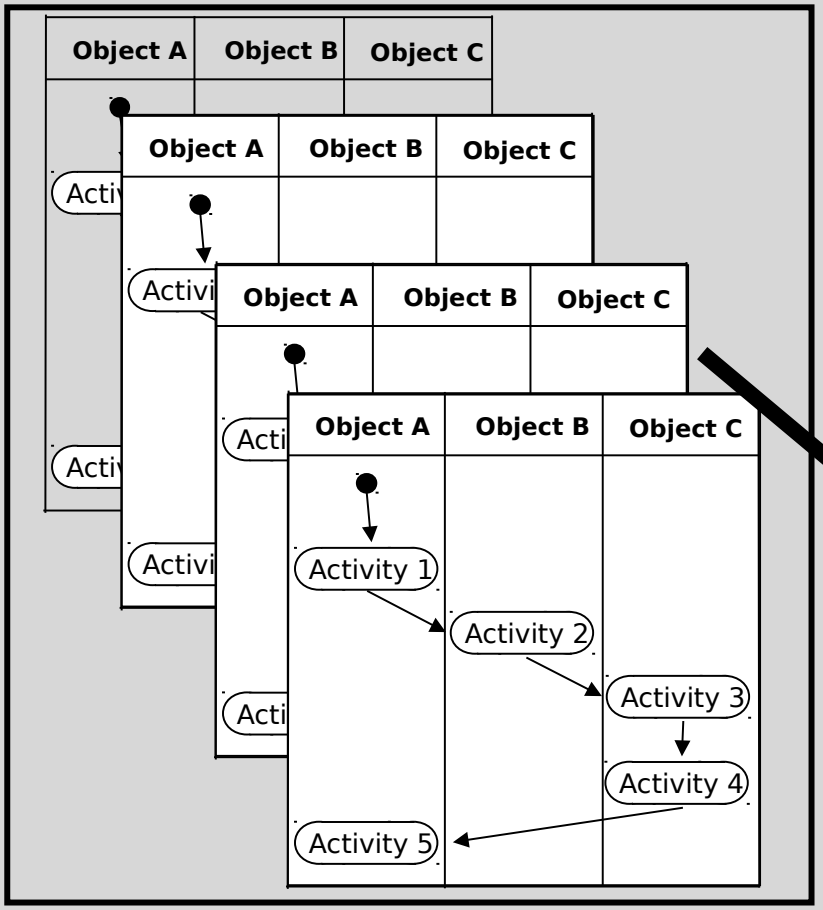
Relationship of Executable Modeling to Verification and Validation



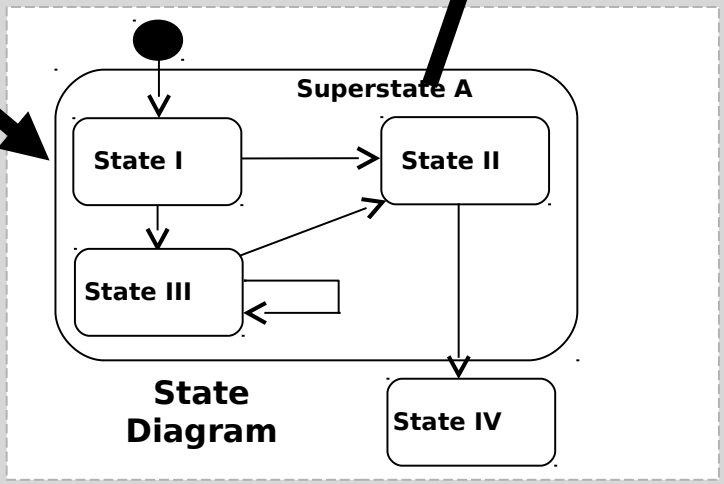
The Role of Executable Modeling is to Verify the Requirements for Consistency
And Validate that the requirements meet the requirements from the level above

BOEING®

Transitioning From Functional Modeling to Executable Modeling



A State Model Is Developed for Each Class



AV Class
Operation 1
Operation 2
Operation 3
.....
Operation n

Bring together Physical, Functional, Data to Evaluate Logical Consistency & Performance

BOEING®

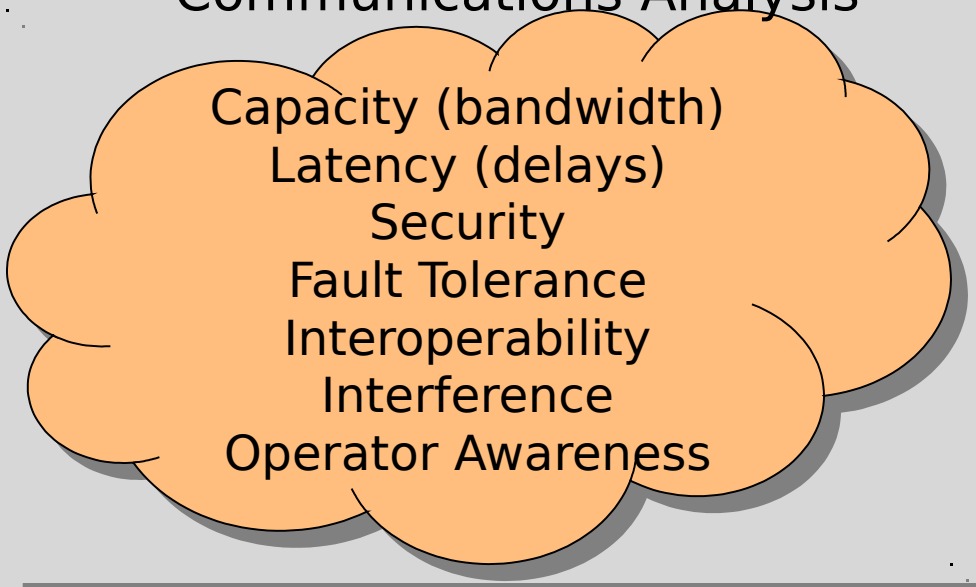
System Analysis & Tool Suitability

Context: Force Structure

Concept: Weapon System

Logical: Segment

Communications Analysis



Context Level Architecture Analy
- Concept Modeled As A “Black Box”
- Model of Domain Elements as Sour
  And Sinks

Concept Level Architecture Analy
- Segment Elements Modeled
  As “Black Boxes”
- Domain Elements Modeled as Sour
  And Sinks

Logical Level Architecture Analys
- Physical Elements Modeled
  As “Black Boxes”
- Concept Elements Modeled as Sou
  And Sinks

Tools Should Be Chosen to Best Fit The Needs of the Purpose of the Analysis

BOEING®

Using UML In The System Engineering Process

Design Of Highly Integrated Systems Require Integrated SE Processes

Adoption of a UML based Object Oriented Architecture Centric SE Process That Uses Modeling and Simulation As Its Core, Meets That Need

- **Best Solution for Behavioral Requirements Development**
  - Strongly and clearly driven by concepts of use and documented knowledge of required interfaces with other systems.
  - Use case based approach fosters multi-discipline participation
- **Provides Appropriate (Use Oriented) Focal Point to Organize and View Other Model Artifacts**
- **Provides Clear Traceability of Derived Behavioral Requirements**
  - Saves $$$ for changes and supports spiral development
- **Provides Strong Foundation for Engineering & Design Processes Automation**
  - Large industry investment and support from Software Engineering

A Disciplined Architecture Centric Systems Engineering Approach Will be Required to Provide Interoperable C4ISR Systems to Meet The Challenges of Information Dominance

BOEING®